Numerical Solution of First Order Initial Value Problem using 7-stage Tenth Order Gauss-Kronrod-Lobatto IIIA Method

Teh Yuan Ying[a] and Nazeeruddin Yaacob[b]

[a]*School of Quantitative Sciences, College of Arts and Sciences, Universiti Utara Malaysia, 06010 UUM Sintok, Kedah DA, Malaysia*
[b]*Department of Mathematics, Faculty of Science, Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor DT, Malaysia*

Abstract. In this paper, a new implicit Runge-Kutta method which based on a 7-point Gauss-Kronrod-Lobatto quadrature formula is developed. The resulting implicit method is a 7-stage tenth order Gauss-Kronrod-Lobatto IIIA method, or in brief as GKLM(7,10)-IIIA. GKLM(7,10)-IIIA requires seven function of evaluations at each integration step and it gives accuracy of order ten. In addition, GKLM(7,10)-IIIA has stage order seven and being A-stable. Numerical experiments compare the accuracy between GKLM(7,10)-IIIA and the classical 5-stage tenth order Gauss-Legendre method in solving some test problems. Numerical results reveal that GKLM(7,10)-IIIA is more accurate than the 5-stage tenth order Gauss-Legendre method because GKLM(7,10)-IIIA has higher stage order.

Keywords: Initial value problem, Gauss-Kronrod-Lobatto quadrature formula, Gauss-Kronrod-Lobatto IIIA method
PACS: 02.60Lj

INTRODUCTION

One-step Runge-Kutta method which is a self-starting numerical method gains tremendous popularity for the computations of numerical solution of first order initial value problems given by

$$y' = f(x, y), \ y(x_0) = \eta. \tag{1}$$

According to Alexander [1-2], the rationale behind the Runge-Kutta method is to approximate the integral in

$$y(x_{n+1}) = y(x_n) + \int_{x_n}^{x_{n+1}} f(x, y(x))dx, \tag{2}$$

by a quadrature formula as follows:

$$y_{n+1} = y_n + h\sum_{i=1}^{s} b_i f(x_n + c_i h, Y_i), \tag{3}$$

where the numbers $b_1, \ldots, b_s$ and $c_1, \ldots, c_s$ which are independent of the function f, are called the quadrature weights and nodes, respectively. The functions Y_i are the stage values which are the approximations to $y(x_n + c_i h)$, $i = 1, \ldots, s$, computed by some other quadrature formulae on the intervals $[x_n, x_n + c_i h]$ as follows:

$$Y_i = y_n + h\sum_{j=1}^{s} a_{ij} f(x_n + c_j h, Y_j), \ i = 1, \ldots, s. \tag{4}$$

If in (4), we have that $a_{ij} = 0$ for $j \geq i$, $i = 1(1)s$, then Y_i is said to be defined explicitly so that formulae in (3) and (4) form an explicit Runge-Kutta method. In most cases, explicit Runge-Kutta method is preferable because it

Proceedings of the 20th National Symposium on Mathematical Sciences
AIP Conf. Proc. 1522, 183-191 (2013); doi: 10.1063/1.4801122
© 2013 AIP Publishing LLC 978-0-7354-1150-0/$30.00

allows explicit stage-by-stage implementation which is very easy to programme using computer. However, numerical analysts also aware that the computational costs involving function evaluations increases rapidly as higher order requirements are imposed [3]. Another disadvantage of explicit Runge-Kutta method is that it has relatively small interval of absolute stability renders them unsuitable for stiff initial value problems [4]. In view of this, we are thus taking interest in implicit Runge-Kutta method. Implicit Runge-Kutta method is defined by the formulae in (3) and (4) when Y_i are defined by a set of s implicit equations. In an implicit Runge-Kutta method, the explicit stage-by-stage implementation scheme enjoyed by explicit Runge-Kutta method is no longer available and needs to be replaced by an iterative computation [5]. Other than this computational difficulty, implicit Runge-Kutta method is an appealing method where higher accuracy can be obtained with fewer function evaluations than for explicit Runge-Kutta method, and it has relatively bigger interval of absolute stability. For excellent surveys and perspectives on these classical methods, see the texts of Hall and Watt [3], Fatunla [4], Butcher [5, 8, 11], Dekker and Verwer [6], Jain [7], Hairer and Wanner [9, 14], Lambert [10], Hairer et al. [12], and Pinto et al. [13].

Throughout the years, the major issue in the implementation of implicit Runge-Kutta method is to find some effective ways in evaluating the stages in the method. One approach taken is to devise some forms of semi-implicit methods so that the stages can be computed sequentially rather than as one great implicit system. These methods are best known as diagonally implicit Runge-Kutta methods, and were widely discussed in Alexander [1-2], Fatunla [4], Butcher [5], Dekker and Verwer [6], Hairer and Wanner [9], Hairer et al. [12], Kværnø et al. [15], Kristensen et al. [16], and Kulikov and Shindin [17]. Another approach discussed in Fatunla [4], Butcher [5, 18], Dekker and Verwer [6], Hairer and Wanner [9], Butcher and Cash [19], Butcher and Chartier [20], Butcher and Chen [21], and Butcher and Wright [22], are called singly implicit Runge-Kutta methods i.e. fully implicit Runge-Kutta methods which possess the matrix (a_{ij}) as in equation (4) with a one point spectrum $\sigma(A)=\{\lambda\}$, with the value of λ chosen for good stability. Some of the researchers even considered parallel Runge-Kutta methods and parallel iterated Runge-Kutta methods as reported in Hairer et al. [12], and Franco and Gomez [23]. The aim of this approach is to speed up the computations for very large problem, for very costly functions, or for fast real-time simulations.

Today, high speed computers are available. In the authors' opinion, fully implicit Runge-Kutta methods should be considered as highly valuable methods. To our best knowledge, there are very few fully implicit Runge-Kutta methods that are based on quadrature formula. Notable examples of fully implicit Runge-Kutta methods with high order and strong stability requirements are the classical Gauss-Legendre methods, families of Radau IA and IIA; and families of Lobatto IIIA, IIIB and IIIC. Therefore it is natural to ask whether we can devise other types of quadrature formulae in order to develop some new implicit Runge-Kutta methods that are more accurate than the Gauss-Legendre type implicit Runge-Kutta methods mentioned above. In view of this, we have developed an implicit Runge-Kutta method that based on the Gauss-Kronrod-Lobatto quadrature formula for the numerical solution of initial value problem in (1).

An n-point Gauss-Lobatto quadrature rule for the integral

$$I(f)=\int_a^b f(x)dx, \tag{5}$$

is a formula of the form

$$G_n f=\sum_{k=1}^{n} w_k f(x_k). \tag{6}$$

with the nodes $a=x_1<x_2<x_3<\cdots<x_n=b$ and positive weights w_k are chosen so that $G_n f=I(f)$, $\forall f\in \mathrm{P}_{2n-3}$ where P_{2n-3} denotes the set of polynomials of degree $2n-3$ [24-25]. The associated Gauss-Kronrod-Lobatto quadrature formula is given by

$$K_{2n-1}f=\sum_{k=1}^{n}\hat{w}_k f(\hat{x}_k)+\sum_{k=1}^{n-1}\tilde{w}_k f(\tilde{x}_k), \tag{7}$$

where $\{\hat{x}_k = x_k\}$ are precisely the one used in equation (6), while all the other $3n-2$ parameters $\{\hat{w}_k\}$, $\{\tilde{w}_k\}$ and $\{\tilde{x}_k\}$ are chosen in such a way that $K_{2n-1}f = I(f)$, $\forall f \in P_{3n-2}$ [26]. According to Calvetti et al. [26], the nodes in the Gauss-Kronrod-Lobatto quadrature formula are ordered so that the following interlacing property is satisfied:

$$a = x_1 < \tilde{x}_1 < x_2 < \tilde{x}_2 < x_3 < \tilde{x}_3 < \cdots < x_{n-1} < \tilde{x}_{n-1} < x_n = b.$$

This paper is organized as follows. Section 2 presents the development of a 7-stage tenth order Gauss-Kronrod-Lobatto IIIA method, or in brief as GKLM(7,10)-IIIA. Numerical comparisons between GKLM(7,10)-IIIA and the 5-stage tenth order Gauss-Legendre method are presented in Section 3. Lastly, some discussions and conclusions are given in Section 4.

7-STAGE TENTH ORDER GAUSS-KRONROD-LOBATTO IIIA METHOD

In this section, we developed an implicit Runge-Kutta method based on 7-point Gauss-Kronrod-Lobatto quadrature formula which consists of four fixed nodes from the 4-point Gauss-Lobatto quadrature formula, and 3 additional nodes. The weights and nodes of a 4-point Gauss-Lobatto quadrature formula are well known, and these values are given by [5]

$$\left\{ w_1 = \frac{1}{12}, w_2 = \frac{5}{12}, w_3 = \frac{5}{12}, w_4 = \frac{1}{12}, x_1 = 0, x_2 = \frac{5-\sqrt{5}}{10}, x_3 = \frac{5+\sqrt{5}}{10}, x_4 = 1 \right\}. \tag{8}$$

The weights of a 4-point Gauss-Lobatto quadrature formula as shown in (8) will not be reused in constructing a 7-point Gauss-Kronrod-Lobatto quadrature formula. For the derivation of the 7-point Gauss-Kronrod-Lobatto quadrature formula, we considered the following function given by

$$f(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5 + a_6x^6 + a_7x^7 + a_8x^8 + a_9x^9. \tag{9}$$

On substituting equations (5), (7) and (9) into $K_{2n-1}f = I(f)$, we obtain the following equation:

$$\int_0^1 (a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5 + a_6x^6 + a_7x^7 + a_8x^8 + a_9x^9)dx = \sum_{k=1}^{4} \hat{w}_k f(\hat{x}_k) + \sum_{k=1}^{3} \tilde{w}_k f(\tilde{x}_k), \tag{10}$$

with $n = 4$, $a = 0$ and $b = 1$. The integration of integral in (10) yields the following result:

$$\begin{aligned} &\int_0^1 (a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5 + a_6x^6 + a_7x^7 + a_8x^8 + a_9x^9)dx \\ &\qquad = a_0 + \frac{a_1}{2} + \frac{a_2}{3} + \frac{a_3}{4} + \frac{a_4}{5} + \frac{a_5}{6} + \frac{a_6}{7} + \frac{a_7}{8} + \frac{a_8}{9} + \frac{a_9}{10}. \end{aligned} \tag{11}$$

On substituting the result in (11) and $\left\{ x_1 = \hat{x}_1 = 0, x_2 = \hat{x}_2 = \frac{5-\sqrt{5}}{10}, x_3 = \hat{x}_3 = \frac{5+\sqrt{5}}{10}, x_4 = \hat{x}_4 = 1 \right\}$ into (10), we obtain the following equation:

$$\begin{aligned} &\hat{w}_1 f(0) + \tilde{w}_1 f(\tilde{x}_1) + \hat{w}_2 f\left(\frac{5-\sqrt{5}}{10}\right) + \tilde{w}_2 f(\tilde{x}_2) + \hat{w}_3 f\left(\frac{5+\sqrt{5}}{10}\right) + \tilde{w}_3 f(\tilde{x}_3) + \hat{w}_4 f(1) \\ &\qquad = a_0 + \frac{a_1}{2} + \frac{a_2}{3} + \frac{a_3}{4} + \frac{a_4}{5} + \frac{a_5}{6} + \frac{a_6}{7} + \frac{a_7}{8} + \frac{a_8}{9} + \frac{a_9}{10}. \end{aligned} \tag{12}$$

Finally, we have to rearrange the left-hand side of (12) in terms of a_i for $i = 0(1)9$ and match these coefficients a_i with those on the right-hand side of (12) in order to obtain a system of ten equations. On solving these ten equations simultaneously using *MATHEMATICA 5.0*, we have obtained the following weights and quadrature nodes of a 7-point Gauss-Kronrod-Lobatto quadrature formula:

$$\left\{\begin{array}{l} \hat{w}_1 = \dfrac{11}{420}, \tilde{w}_1 = \dfrac{36}{245}, \hat{w}_2 = \dfrac{125}{588}, \tilde{w}_2 = \dfrac{8}{35}, \hat{w}_3 = \dfrac{125}{588}, \tilde{w}_3 = \dfrac{36}{245}, \hat{w}_4 = \dfrac{11}{420}, \\ x_1 = 0, \tilde{x}_1 = \dfrac{3-\sqrt{6}}{6}, x_2 = \dfrac{5-\sqrt{5}}{10}, \tilde{x}_2 = \dfrac{1}{2}, x_3 = \dfrac{5+\sqrt{5}}{10} \tilde{x}_3 = \dfrac{3+\sqrt{6}}{6}, x_4 = 1 \end{array}\right\}, \tag{13}$$

or, in the sense of the weights and abscissae of an implicit Runge-Kutta method is

$$\left\{\begin{array}{l} b_1 = \dfrac{11}{420}, b_2 = \dfrac{36}{245}, b_3 = \dfrac{125}{588}, b_4 = \dfrac{8}{35}, b_5 = \dfrac{125}{588}, b_6 = \dfrac{36}{245}, b_7 = \dfrac{11}{420}, \\ c_1 = 0, c_2 = \dfrac{3-\sqrt{6}}{6}, c_3 = \dfrac{5-\sqrt{5}}{10}, c_4 = \dfrac{1}{2}, c_5 = \dfrac{5+\sqrt{5}}{10} c_6 = \dfrac{3+\sqrt{6}}{6}, c_7 = 1 \end{array}\right\}. \tag{14}$$

In order to construct the implicit Runge-Kutta method based on 7-point Gauss-Kronrod-Lobatto quadrature formula, the choice of a_{ij} for $i, j = 1(1)7$ is to satisfy all the 49 order conditions of [5, 9]

$$C(7) = \sum_{j=1}^{7} a_{ij} c_j^{k-1} = \frac{c_i^k}{k} \text{ for } i = 1(1)7 \text{ and } k = 1(1)7.$$

On solving these 49 equations simultaneously using *MATHEMATICA 5.0* yields the solution of the parameters a_{ij} for $i, j = 1(1)7$ as shown below:

$$\begin{cases} a_{11} = 0, a_{12} = 0, a_{13} = 0, a_{14} = 0, a_{15} = 0, a_{16} = 0, a_{17} = 0, a_{21} = \dfrac{1877+96\sqrt{6}}{60480}, a_{22} = \dfrac{2592-109\sqrt{6}}{35280}, \\ a_{23} = \dfrac{25\left(360-\sqrt{5\left(30769+448\sqrt{30}\right)}\right)}{84672}, a_{24} = \dfrac{108-41\sqrt{6}}{945}, a_{25} = \dfrac{25\left(360-\sqrt{5\left(30769-448\sqrt{30}\right)}\right)}{84672}, \\ a_{26} = \dfrac{2592-1019\sqrt{6}}{35280}, a_{27} = \dfrac{-293+96\sqrt{6}}{60480}, a_{31} = \dfrac{2425-61\sqrt{5}}{105000}, a_{32} = \dfrac{6\left(375+\sqrt{30\left(6149+140\sqrt{30}\right)}\right)}{30625}, \\ a_{33} = \dfrac{625-\sqrt{5}}{5880}, a_{34} = \dfrac{4}{35} - \dfrac{264}{875\sqrt{5}}, a_{35} = \dfrac{625-253\sqrt{5}}{5880}, a_{36} = \dfrac{6\left(-375+\sqrt{30\left(6149-140\sqrt{30}\right)}\right)}{30625}, \\ a_{37} = \dfrac{325-61\sqrt{5}}{105000}, a_{41} = \dfrac{193}{6720}, a_{42} = \dfrac{3\left(96+35\sqrt{6}\right)}{3920}, a_{43} = \dfrac{25\left(40+21\sqrt{5}\right)}{9408}, a_{44} = \dfrac{4}{35}, a_{45} = \dfrac{25\left(40-21\sqrt{5}\right)}{9408}, \\ a_{46} = \dfrac{3\left(96-35\sqrt{6}\right)}{3920}, a_{47} = -\dfrac{17}{6720}, a_{51} = \dfrac{2425+61\sqrt{5}}{105000}, a_{52} = \dfrac{6\left(375+\sqrt{30\left(6149-140\sqrt{30}\right)}\right)}{30625}, \\ a_{53} = \dfrac{625+253\sqrt{5}}{5800}, a_{54} = \dfrac{4\left(125+66\sqrt{5}\right)}{4375}, a_{55} = \dfrac{625+\sqrt{5}}{5880}, a_{56} = \dfrac{6\left(375-\sqrt{30\left(6149+140\sqrt{30}\right)}\right)}{30625}, \end{cases}$$

$$a_{57}=\frac{325+61\sqrt{5}}{105000},a_{61}=\frac{1877-96\sqrt{6}}{60480},a_{62}=\frac{2592+1019\sqrt{6}}{35280},a_{63}=\frac{25\left(360+\sqrt{5\left(30769-448\sqrt{30}\right)}\right)}{84672},$$
$$a_{64}=\frac{108+41\sqrt{6}}{945},a_{65}=\frac{25\left(360+\sqrt{5\left(30769+448\sqrt{30}\right)}\right)}{84672},a_{66}=\frac{2592+109\sqrt{6}}{35280},a_{67}=\frac{-293-96\sqrt{6}}{60480},$$
$$\left.a_{71}=\frac{11}{420},a_{72}=\frac{36}{245},a_{73}=\frac{125}{588},a_{74}=\frac{8}{35},a_{75}=\frac{125}{588},a_{76}=\frac{36}{245},a_{77}=\frac{11}{420}\right\}. \tag{15}$$

On substituting the values in (14) and (15) with $s=7$ into equations (3) and (4), we obtained the 7-stage tenth order Gauss-Kronrod-Lobatto IIIA method, or in brief as GKLM(7,10)-IIIA. GKLM(7,10)-IIIA has proved to possess tenth order of accuracy because the parameters in (14) satisfy all the order conditions in [5, 9]

$$B(10)=\sum_{i=1}^{7}b_i c_i^{k-1}=\frac{1}{k}\ \text{for}\ k=1(1)10.$$

In addition, the parameters in (14) and (15) also satisfy all the order conditions in [5, 9]

$$D(3)=\sum_{i=1}^{7}b_i c_i^{k-1}a_{ij}=\frac{b_j}{k}\left(1-c_j^k\right)\ \text{for}\ j=1(1)7\ \text{and}\ k=1(1)3.$$

We note that $C(7)$, $B(10)$ and $D(3)$ are simplifying assumptions that are being used to relate the parameters a_{ij}, b_i and c_i of GKLM(7,10)-IIIA. Since GKLM(7,10)-IIIA satisfies $C(7)$, then we can claim that GKLM(7,10)-IIIA has stage order 7.

The stability function of a Runge-Kutta method can be obtained by utilizing it to the Dahlquist's test equation $y'=\lambda y$, $\mathrm{Re}(\lambda)<0$, which yields a difference equation given by $y_{n+1}=R(z)y_n$ where $R(z)$ is the stability function of a Runge-Kutta method with $z=h\lambda$. On the other hand, Dekker and Verwer [6] has developed an alternative form for the stability function of a Runge-Kutta method given by

$$R(z)=\frac{\det\left[\mathbf{I}-z\mathbf{A}+\mathbf{c}\mathbf{b}^{T}\right]}{\det\left[\mathbf{I}-z\mathbf{A}\right]}. \tag{16}$$

where in the case of a 7-stage Runge-Kutta method, $\mathbf{I}$ is a 7×7 identity matrix, $\mathbf{A}$ is a matrix containing the elements a_{ij} for $i,j=1(1)7$, $c=(1\ \ 1\ \ 1\ \ 1\ \ 1\ \ 1\ \ 1)^{T}$ and $\mathbf{b}$ is a row vector containing the elements b_i for $i=1(1)7$. Upon these substitutions from (14) and (15), the stability function for GKLM(7,10)-IIIA is given by

$$R(z)=\frac{604800+302400z+68880z^2+9240z^3+780z^4+40z^5+z^6}{604800-302400z+68880z^2-9240z^3+780z^4-40z^5+z^6}. \tag{17}$$

By taking $z=x+iy$ in (17), we obtain a plot of the stability function (17) as shown in Figure 1. The shaded region in Figure 1 is the region of absolute stability of GKLM(7,10)-IIIA where the condition $|R(z)|\le 1$ is satisfied.

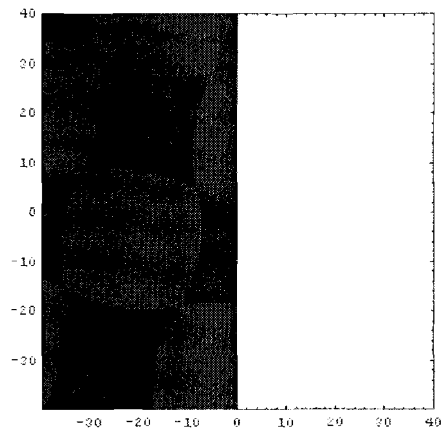

FIGURE 1. Stabilty region of GKLM(7,10)-IIIA

Observed that the region of absolute stability of GKLM(7,10)-IIIA contains the whole left-half complex plane, which suggest that GKLM(7,10)-IIIA is A-stable. However, it is not L-stable because $|R(z)| \to 1$ as $\mathrm{Re}(z) \to -\infty$.

NUMERICAL EXPERIMENTS AND COMPARISONS

In this section, some test problems are used to check the accuracy of GKLM(7,10)-IIIA using different number of integration steps. We present the maximum absolute errors over the integration interval given by $\max_{0 \le n \le N} \{ |y(x_n) - y_n| \}$ where N is the number of integration steps. We note that $y(x_n)$ and y_n represent the theoretical and numerical solutions of a test problem at point x_n, respectively. The numerical results obtained from GKLM(7,10)-IIIA are compared with the numerical results obtained from the 5-stage tenth order Gauss-Legendre method. The 5-stage tenth order Gauss-Legendre method consists of the formulae in (3) and (4) with the following parameters:

$$\begin{cases} b_1 = b_5 = \dfrac{322 - 13\sqrt{70}}{1800}, b_2 = b_4 = \dfrac{322 + 13\sqrt{70}}{1800}, b_3 = \dfrac{64}{225}, c_1 = \dfrac{1}{2} - \dfrac{1}{6}\sqrt{5 + 2\sqrt{\dfrac{10}{7}}}, c_2 = \dfrac{1}{2} - \dfrac{1}{6}\sqrt{5 - 2\sqrt{\dfrac{10}{7}}}, c_3 = \dfrac{1}{2}, \\ c_4 = \dfrac{1}{2} + \dfrac{1}{6}\sqrt{5 - 2\sqrt{\dfrac{10}{7}}}, c_5 = \dfrac{1}{2} + \dfrac{1}{6}\sqrt{5 + 2\sqrt{\dfrac{10}{7}}}, a_{11} = a_{55} = \dfrac{322 - 13\sqrt{70}}{3600}, a_{13} = \dfrac{32}{225} - \dfrac{4}{1215}\sqrt{2075 - 412\sqrt{\dfrac{10}{7}}}, \\ a_{15} = \dfrac{2254 - 91\sqrt{70} - 30\sqrt{5390 - 364\sqrt{70}}}{25200}, a_{22} = a_{44} = \dfrac{322 + 13\sqrt{70}}{3600}, a_{23} = \dfrac{32}{225} - \dfrac{4}{1215}\sqrt{2075 + 412\sqrt{\dfrac{10}{7}}}, \\ a_{24} = \dfrac{2254 + 91\sqrt{70} - 30\sqrt{5390 + 364\sqrt{70}}}{25200}, a_{31} = \dfrac{2576 - 104\sqrt{70} + 15\sqrt{6\left(6965 - 614\sqrt{70}\right)}}{2880}, \\ a_{32} = \dfrac{2576 + 104\sqrt{70} + 15\sqrt{6\left(6965 + 614\sqrt{70}\right)}}{2880}, a_{33} = \dfrac{32}{225}, a_{34} = \dfrac{2576 + 104\sqrt{70} - 15\sqrt{6\left(6965 + 614\sqrt{70}\right)}}{2880}, \\ a_{35} = \dfrac{2576 - 104\sqrt{70} - 15\sqrt{6\left(6965 - 614\sqrt{70}\right)}}{2880}, a_{42} = \dfrac{2254 + 91\sqrt{70} + 30\sqrt{5390 + 364\sqrt{70}}}{25200}, \\ a_{43} = \dfrac{32}{225} + \dfrac{4}{1215}\sqrt{2075 + 412\sqrt{\dfrac{10}{7}}}, a_{51} = \dfrac{2254 - 91\sqrt{70} + 30\sqrt{5390 - 364\sqrt{70}}}{25200}, a_{53} = \dfrac{32}{225} + \dfrac{4}{1215}\sqrt{2075 - 412\sqrt{\dfrac{10}{7}}}, \end{cases}$$

Downloaded 26 Jun 2013 to 163.5...26. This article is copyrighted as indicated in the abstract. Reuse of AIP content is subject to the terms at: ...

$$a_{12} = \frac{-1323 + 945\sqrt{70} + 210\sqrt{350 - 20\sqrt{70}} - 285\sqrt{7\left(35 - 2\sqrt{70}\right)} - 215\sqrt{7\left(35 + 2\sqrt{70}\right)} - 140\sqrt{350 + 20\sqrt{70}}}{12600\left(-4 + \sqrt{70}\right)},$$

$$a_{14} = \frac{-1323 + 945\sqrt{70} - 210\sqrt{350 - 20\sqrt{70}} + 285\sqrt{7\left(35 - 2\sqrt{70}\right)} - 215\sqrt{7\left(35 + 2\sqrt{70}\right)} - 140\sqrt{350 + 20\sqrt{70}}}{12600\left(-4 + \sqrt{70}\right)},$$

$$a_{21} = \frac{-140\sqrt{350 - 20\sqrt{70}} + 215\sqrt{7\left(35 - 2\sqrt{70}\right)} + 3\left(441 + 315\sqrt{70} + 95\sqrt{7\left(35 + 2\sqrt{70}\right)} + 70\sqrt{350 + 20\sqrt{70}}\right)}{\left(12600\left(4 + \sqrt{70}\right)\right)},$$

$$a_{25} = \frac{-140\sqrt{350 - 20\sqrt{70}} + 215\sqrt{7\left(35 - 2\sqrt{70}\right)} + 3\left(441 + 315\sqrt{70} - 95\sqrt{7\left(35 + 2\sqrt{70}\right)} - 70\sqrt{350 + 20\sqrt{70}}\right)}{12600\left(4 + \sqrt{70}\right)},$$

$$a_{41} = \frac{140\sqrt{350 - 20\sqrt{70}} - 215\sqrt{7\left(35 - 2\sqrt{70}\right)} + 3\left(441 + 315\sqrt{70} + 95\sqrt{7\left(35 + 2\sqrt{70}\right)} + 70\sqrt{350 + 20\sqrt{70}}\right)}{12600\left(4 + \sqrt{70}\right)},$$

$$a_{45} = \frac{140\sqrt{350 - 20\sqrt{70}} - 215\sqrt{7\left(35 - 2\sqrt{70}\right)} + 3\left(441 + 315\sqrt{70} - 95\sqrt{7\left(35 + 2\sqrt{70}\right)} - 70\sqrt{350 + 20\sqrt{70}}\right)}{12600\left(4 + \sqrt{70}\right)},$$

$$a_{52} = \frac{-1323 + 945\sqrt{70} + 210\sqrt{350 - 20\sqrt{70}} - 285\sqrt{7\left(35 - 2\sqrt{70}\right)} + 215\sqrt{7\left(35 + 2\sqrt{70}\right)} + 140\sqrt{350 + 20\sqrt{70}}}{12600\left(-4 + \sqrt{70}\right)},$$

$$\left. a_{54} = \frac{-1323 + 945\sqrt{70} - 210\sqrt{350 - 20\sqrt{70}} + 285\sqrt{7\left(35 - 2\sqrt{70}\right)} + 215\sqrt{7\left(35 + 2\sqrt{70}\right)} + 140\sqrt{350 + 20\sqrt{70}}}{12600\left(-4 + \sqrt{70}\right)} \right\}.$$

Problem 1 (Ramos [27])

$$y'(x) = -100y(x) + 99e^{2x}, \quad y(0) = 0, \quad x \in [0, 10].$$

The theoretical solution is given by $y(x) = 33/34(e^{2x} - e^{-100x})$.

TABLE (1). Maximum absolute errors with respect to number of integration steps, N (*Problem 1*)

N	5-stage tenth order Gauss-Legendre method	GKLM(7,10)-IIIA
100	1.13687(-02)	8.14525(-04)
160	2.54095(-04)	3.97925(-05)
200	5.60818(-05)	7.72878(-06)
300	2.51315(-06)	1.07288(-06)
320	1.47579(-06)	1.78814(-07)

Problem 2 (Yaakub and Evans [28])

$$y''(x) + 101y'(x) + 100y(x) = 0, \quad y(0) = 1.01, \quad y'(0) = -2, \quad x \in [0, 10].$$

The theoretical solution is given by $y(x)=0.01e^{-100x}+e^{-x}$. *Problem 2* can also be written as a system, i.e.

$$y_1'(x)=y_2(x),\ y_1(0)=1.01,\ x\in[0,10],$$
$$y_2'(x)=-100y_1(x)-101y_2(x),\ y_2(x)=-2,\ x\in[0,10].$$

The theoretical solutions of this system are given by $y_1(x)=y(x)=0.01e^{-100x}+e^{-x}$, $y_2(x)=y'(x)=-e^{-100x}-e^{-x}$.

TABLE (2). Maximum absolute errors with respect to number of integration steps, N (*Problem 2*)

N	5-stage tenth order Gauss-Legendre method	GKLM(7,10)-IIIA
100	3.75398(-05)	8.39208(-06)
160	2.61795(-06)	1.74751(-06)
200	5.77812(-07)	7.96299(-08)
300	2.58931(-08)	3.04464(-09)
320	1.52051(-08)	5.07516(-09)

CONCLUSIONS

In this paper, we have developed a 7-stage tenth order Gauss-Kronrod-Lobatto IIIA method, or in brief as GKLM(7,10)-IIIA. GKLM(7,10)-IIIA possesses stage order 7 and A-stable. From Tables 1 and 2, we can see that GKLM(7,10)-IIIA with stage order 7 is more accurate than 5-stage tenth order Gauss-Legendre method with stage order 5, in solving both *Problem 1* and *Problem 2* for $N=100$, 160, 200, 300 and 320. Therefore, it is evident that implicit Runge-Kutta method with higher stage order gives more accurate numerical solution for the first order initial value problem in (1).

REFERENCES

1. R. Alexander, *SIAM J. Numer. Anal.* **14**, 1006–1021 (1977).
2. R. Alexander, *Appl. Numer. Math.* **46**, 1–17 (2003).
3. G. Hall and J. M. Watt, *Modern Numerical Methods for Ordinary Differential Equations*, London: Oxford University Press, 1976.
4. S. O. Fatunla, *Numerical Methods for Initial Value Problems in Ordinary Differential Equations*, San Diego: Academic Press Inc., 1988.
5. J. C. Butcher, *Numerical Methods for Ordinary Differential Equations*, West Sussex: John Wiley & Sons Ltd., 2003.
6. K. Dekker and J. G. Verwer, *Stability of Runge-Kutta Methods for Stiff Nonlinear Differential Equations*, Amsterdam: Elsevier Science Publishers, 1984.
7. M. K. Jain, *Numerical Solution of Differential Equations*, New Dehli: Wiley Eastern Limited, 1984.
8. J. C. Butcher, *The Numerical Analysis of Ordinary Differential Equations*, Chichester: John Wiley & Sons Ltd., 1987.
9. E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II*, Berlin: Springer-Verlag, 1991.
10. J. D. Lambert, *Numerical Methods for Ordinary Differential Systems*, Chichester: John Wiley & Sons Ltd., 1991.
11. J. C. Buether, *J. Comput. Appl. Math.* **43**, 231–242 (1992).
12. E. Hairer, S. P. Nørsett and G. Wanner, *Solving Ordinary Differential Equations I*, Berlin: Speinger-Verlag, 1993.
13. S. G. Pinto, S. P. Rodriguez and J. I. M. Toreal, *J. Comput. Appl. Math.* **82**, 129–148 (1997).
14. E. Hairer and G. Wanner, *J. Comput. Appl. Math.* **111**, 93–111 (1999).
15. A. Kværnø, S. P. Nørsett and B. Owren, *Appl. Numer. Math.* **22**, 263–277 (1996).
16. M. R. Kristensen, J. B. Jorgensen, P. G. Thomsen and S. B. Jorgensen, *Comput. Chem. Eng.* **28**, 2695–2707 (2004).
17. G. Y. Kulikov and S. K. Shindin, *Appl. Numer. Math.* **59**, 707–722 (2008).
18. J. C. Butcher, *J. ACM.* **26**, 731–738 (1979).
19. J. C. Butcher and J. R. Cash, *SIAM J. Numer. Anal.* **27**, 753–761 (1990).
20. J. C. Butcher and P. Chartier, *Appl. Numer. Math.* **24**, 343–350 (1997).
21. J. C. Butcher and D. J. L. Chen, *Appl. Numer. Math.* **28**, 193–207 (1998).
22. J. C. Butcher and W. M. Wright, *Appl. Numer. Math.* **56**, 358–373 (2006).
23. J. M. Franco and I. Gomez, *J. Comput. Appl. Math.* **87**, 119–134 (1997).
24. P. J. Davis and P. Rabinowitz, *Methods of Numerical Integration*, Orlando: Academic Press Inc., 1984.
25. P. K. Kythe and M. R. Schäferkotter, *Handbook of Computational Methods for Integration*, Boca Raton: Chapman & Hall/CRC, 2005.

26. D. Calvetti, G. H. Golub, W. B. Gragg and L. Reichel, *Math. Comput.* **69**, 1035–1052 (2000).
27. H. Ramos, *Appl. Math. Comput.* **189**, 710–718 (2007).
28. A. R. Yaakub and D. J. Evans, *Int. J. Comp. Math.* **80**, 95–104 (2003).